



Showing that the race model inequality is not violated

Gondan, Matthias; Riehl, Verena; Blurton, Steven Paul

Published in:
Behavior Research Methods

DOI:
[10.3758/s13428-011-0147-z](https://doi.org/10.3758/s13428-011-0147-z)

Publication date:
2012

Document version
Publisher's PDF, also known as Version of record

Citation for published version (APA):
Gondan, M., Riehl, V., & Blurton, S. P. (2012). Showing that the race model inequality is not violated. *Behavior Research Methods*, 44(1), 248-255. <https://doi.org/10.3758/s13428-011-0147-z>

```

# R script for vertical non-inferiority test (tested with R version 2.13.1)
# Supplemental material for "Showing that the race model inequality is not
# violated"
#
# Hybrid bootstrap confidence interval for maximal distance (Freitag et al.,
# 2006, Eq. 4 with m=n)
# H_0:  $F_R(t) + \delta = F_T(t)$ , for some t
# H_1:  $F_R(t) + \delta > F_T(t)$ , for all t
#
# Arguments
#   dT, dR: samples T and R
#   delta: non-inferiority margin [default 0.1]
#   nboot: number of bootstrap cycles [default 9999]
#
# Return value
#   P: P-value of non-inferiority test

dci = function(dT, dR, delta=0.1, nboot=9999)
{
  # observed distance
  range=union(dT, dR)
  dmax_obs = max(ecdf(dT)(range) - ecdf(dR)(range))

  # bootstrap confidence interval
  m = numeric(nboot)
  for(i in 1:nboot)
  {
    sT = sample(dT, replace=TRUE)
    sR = sample(dR, replace=TRUE)
    m[i] = max(ecdf(sT)(range) - ecdf(sR)(range))
  } # for

  c(P=mean(2*dmax_obs - m > delta))
} # dci

# dci(dT=rnorm(100, 100, 15), dR=rnorm(100, 100, 15)) # only for testing

# Invoke non-inferiority test with data from redundant signals task
# H_0:  $F_{AV}(t) \geq F_A(t) + F_V(t) + \delta$ , for some t
# H_1:  $F_{AV}(t) < F_A(t) + F_V(t) + \delta$ , for all t
#
# Arguments
#   tA, tV, tAV: response times for auditory, visual, audiovisual stimuli
#   delta: vertical non-inferiority margin in units of t [default 10 ms]
#   nboot: number of bootstrap cycles [default 9999]
#
# Return value (see dci)
#   P: one-tailed P-value (< .05 means that race model inequality holds)

vhold = function(tA, tV, tAV, delta=0.1, nboot=9999)
{
  # lower 50% of 1:1 mixture of tA and tV
  dci(tAV, sort(c(tA, tV))[1: floor((length(tA)+length(tV))/2)], delta, nboot)
} # vhold

# only for testing
# vhold(rnorm(100, 100, 15), rnorm(100, 100, 15), pmin(rnorm(100, 100, 15),
#   rnorm(100, 100, 15)))

```

```

# Run the test for data saved in a file
#
# Cond  RT
# A      526
# V      324
# AV     234
# V      Inf # omitted response
#
# Arguments
#   fname: path to results file [forward slashes, default C:/RTDATA/obs01.txt]
#   delta: verical non-inferiority margin [default: 0.1]
#   nboot: number of bootstrap cycles [default: 9999]
#
# Return value (see also dci)
#   P: one-tailed P-value of bootstrap test (< .05 means that race model
#      inequality holds)

rmi.vtest = function(fname='C:/RTDATA/obs01.txt', delta=0.1, nboot=9999)
{
  d = read.table(fname, header=TRUE)
  s = split(d$RT, d$Cond)
  vhold(tA=s$A, tV=s$V, tAV=s$AV, delta, nboot)
} # rmi.vtest

# Uncomment to generate example data in C:\RTDATA\obs01.txt
# set.seed(1234)
# tA = cbind(Cond='A', RT=round(rnorm(60, 100, 15)))
# tV = cbind('V', round(rnorm(80, 100, 15)))
# tAV = cbind('AV', round(pmin(rnorm(100, 100, 15), rnorm(100, 100, 15))))
# write.table(rbind(tA, tV, tAV), 'C:/RTDATA/obs01.txt', quote=FALSE,
#   row.names=FALSE, sep='\t')

# Example use (requires valid results file in C:\RTDATA\obs01.txt)
rmi.vtest(fname='C:/RTDATA/obs01.txt', delta=0.05, nboot=9999)

```